In [2]:
```
import pandas as pd

sql = pd.read_csv(r"C:\Users\santo\OneDrive\Desktop\MY SQL\Dataset_1_202511181007.csv")
sql.head()
```

Out[2]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	Restaura
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	Unmarried partner	...	NaN	
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	Unmarried partner	...	NaN	
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female	21	Unmarried partner	...	NaN	

5 rows × 27 columns

In [27]:
```
print(sql2.columns)
```

```
Index(['destination', 'passanger', 'weather', 'temperature', 'time', 'coupon',
       'expiration', 'gender', 'age', 'maritalStatus', 'has_children',
       'education', 'occupation', 'income', 'car', 'Bar', 'CoffeeHouse',
       'CarryAway', 'RestaurantLessThan20', 'Restaurant20To50',
       'toCoupon_GEQ5min', 'toCoupon_GEQ15min', 'toCoupon_GEQ25min',
       'direction_same', 'direction_opp', 'Y', 'row_count'],
      dtype='object')
```

In [3]:
```
sql[['weather','temperature']]
```

Out[3]:

	weather	temperature
0	Sunny	55
1	Sunny	80
2	Sunny	80
3	Sunny	80
4	Sunny	80
...	...	...
12679	Rainy	55
12680	Rainy	55
12681	Snowy	30
12682	Snowy	30
12683	Sunny	80

12684 rows × 2 columns

In [4]:

```
sql.head(10)
```

Out[4]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	Restaura
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	Unmarried partner	...	NaN	
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	Unmarried partner	...	NaN	
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female	21	Unmarried partner	...	NaN	
5	No Urgent Place	Friend(s)	Sunny	80	6PM	Restaurant(<20)	2h	Female	21	Unmarried partner	...	NaN	
6	No Urgent Place	Friend(s)	Sunny	55	2PM	Carry out & Take away	1d	Female	21	Unmarried partner	...	NaN	
7	No Urgent Place	Kid(s)	Sunny	80	10AM	Restaurant(<20)	2h	Female	21	Unmarried partner	...	NaN	
8	No Urgent Place	Kid(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	Unmarried partner	...	NaN	
9	No Urgent Place	Kid(s)	Sunny	80	10AM	Bar	1d	Female	21	Unmarried partner	...	NaN	

10 rows × 27 columns

In [5]:

```
sql['passanger']
```

Out[5]:

```
0            Alone
1        Friend(s)
2        Friend(s)
3        Friend(s)
4        Friend(s)
           ...    
12679      Partner
12680        Alone
12681        Alone
12682        Alone
12683        Alone
Name: passanger, Length: 12684, dtype: object
```

In [6]:

```
# SELECT*FROM DATASET_1 WHERE DESTINATION-'HOME'
sql[sql['destination']=='Home']
```

Out[6]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	Rest
13	Home	Alone	Sunny	55	6PM	Bar	1d	Female	21	Unmarried partner	...	NaN	
14	Home	Alone	Sunny	55	6PM	Restaurant(20-50)	1d	Female	21	Unmarried partner	...	NaN	
15	Home	Alone	Sunny	80	6PM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
35	Home	Alone	Sunny	55	6PM	Bar	1d	Male	21	Single	...	4~8	
36	Home	Alone	Sunny	55	6PM	Restaurant(20-50)	1d	Male	21	Single	...	4~8	
...	...	...	...	...	...	...	...	...	...	...	...	...	
12675	Home	Alone	Snowy	30	10PM	Coffee House	2h	Male	26	Single	...	1~3	
12676	Home	Alone	Sunny	80	6PM	Restaurant(20-50)	1d	Male	26	Single	...	1~3	
12677	Home	Partner	Sunny	30	6PM	Restaurant(<20)	1d	Male	26	Single	...	1~3	
12678	Home	Partner	Sunny	30	10PM	Restaurant(<20)	2h	Male	26	Single	...	1~3	
12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	1d	Male	26	Single	...	1~3	

3237 rows × 27 columns

In [7]:

```
# SELECT* FROM DATASET_1 ORDERD BY COUPON
sql.sort_values('coupon')
```

Out[7]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	R
11702	Home	Partner	Sunny	30	10PM	Bar	2h	Female	50plus	Married partner	...	4~8	
9930	No Urgent Place	Alone	Snowy	30	2PM	Bar	1d	Female	21	Single	...	gt8	
10632	Home	Alone	Rainy	55	6PM	Bar	1d	Male	21	Single	...	gt8	
7997	No Urgent Place	Friend(s)	Rainy	55	10PM	Bar	2h	Male	26	Unmarried partner	...	4~8	
11166	Work	Alone	Snowy	30	7AM	Bar	1d	Female	41	Married partner	...	gt8	
...	...	...	...	...	...	...	...	...	...	...	...	...	
10476	Home	Alone	Sunny	80	6PM	Restaurant(<20)	1d	Female	31	Unmarried partner	...	1~3	
5447	Home	Alone	Sunny	80	10PM	Restaurant(<20)	2h	Female	50plus	Single	...	less1	
10478	Home	Alone	Snowy	30	10PM	Restaurant(<20)	2h	Female	31	Unmarried partner	...	1~3	
5440	No Urgent Place	Alone	Sunny	80	2PM	Restaurant(<20)	2h	Female	50plus	Single	...	less1	
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	Unmarried partner	...	NaN	

12684 rows × 27 columns

In [8]:

```
sql.rename(columns={'destination':'Destination'},inplace=True)
```

In [9]:

```
sql
```

Out[9]:

	Destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	Rest
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	Unmarried partner	...	NaN	
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	Unmarried partner	...	NaN	
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female	21	Unmarried partner	...	NaN	
...	...	...	...	...	...	...	...	...	...	...	...	...	
12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	1d	Male	26	Single	...	1~3	
12680	Work	Alone	Rainy	55	7AM	Carry out & Take away	1d	Male	26	Single	...	1~3	
12681	Work	Alone	Snowy	30	7AM	Coffee House	1d	Male	26	Single	...	1~3	
12682	Work	Alone	Snowy	30	7AM	Bar	1d	Male	26	Single	...	1~3	
12683	Work	Alone	Sunny	80	7AM	Restaurant(20-50)	2h	Male	26	Single	...	1~3	

12684 rows × 27 columns

In [10]:

```
sql.groupby('occupation').size().to_frame('Count').reset_index()
```

Out[10]:

	occupation	Count
0	Architecture & Engineering	175
1	Arts Design Entertainment Sports & Media	629
2	Building & Grounds Cleaning & Maintenance	44
3	Business & Financial	544
4	Community & Social Services	241
5	Computer & Mathematical	1408
6	Construction & Extraction	154
7	Education&Training&Library	943
8	Farming Fishing & Forestry	43
9	Food Preparation & Serving Related	298
10	Healthcare Practitioners & Technical	244
11	Healthcare Support	242
12	Installation Maintenance & Repair	133
13	Legal	219
14	Life Physical Social Science	170
15	Management	838
16	Office & Administrative Support	639
17	Personal Care & Service	175
18	Production Occupations	110
19	Protective Service	175
20	Retired	495
21	Sales & Related	1093

	occupation	Count
22	Student	1584
23	Transportation & Material Moving	218
24	Unemployed	1870

In [11]:

```
sql.groupby('weather')['temperature'].mean().to_frame('avg_temp').reset_index()
```

Out[11]:

	weather	avg_temp
0	Rainy	55.000000
1	Snowy	30.000000
2	Sunny	68.946271

In [12]:

```
sql.groupby('weather')['temperature'].size().to_frame('Count_temp').reset_index()
```

Out[12]:

	weather	Count_temp
0	Rainy	1210
1	Snowy	1405
2	Sunny	10069

In [13]:

```
sql.groupby('weather')['temperature'].nunique().to_frame('count_distinct_temp').reset_index()
```

Out[13]:

	weather	count_distinct_temp
0	Rainy	1
1	Snowy	1
2	Sunny	3

In [14]:
```
sql.groupby('weather')['temperature'].sum().to_frame('sum_temp').reset_index()
```

Out[14]:

	weather	sum_temp
0	Rainy	66550
1	Snowy	42150
2	Sunny	694220

In [15]:
```
sql.groupby('weather')['temperature'].min().to_frame('min_temp').reset_index()
```

Out[15]:

	weather	min_temp
0	Rainy	55
1	Snowy	30
2	Sunny	30

In [16]:
```
sql.groupby('weather')['temperature'].max().to_frame('max_temp').reset_index()
```

Out[16]:

	weather	max_temp
0	Rainy	55
1	Snowy	30
2	Sunny	80

In [17]:
```
sql.groupby('occupation').filter(lambda x: x['occupation'].iloc[0] ==
'Student').groupby('occupation').size()
```

Out[17]:
```
occupation
Student    1584
dtype: int64
```

In [41]:
```
sql2 = pd.read_csv(r"C:\Users\santo\OneDrive\Desktop\Table_to_join_202511181007.csv")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
Cell In[41], line 1
----> 1 sql2 = pd.read_csv(r"C:\Users\santo\OneDrive\Desktop\Table_to_join_202511181007.csv")

File ~\anaconda3\Lib\site-packages\pandas\io\parsers\readers.py:1026, in read_csv(filepath_or_buffer, sep, delimiter, header, names, index_col, usecols, dtype, engine, converters, true_values, false_values, skipinitialspace, skiprows, skipfooter, nrows, na_values, keep_default_na, na_filter, verbose, skip_blank_lines, parse_dates, infer_datetime_format, keep_date_col, date_parser, date_format, dayfirst, cache_dates, iterator, chunksize, compression, thousands, decimal, lineterminator, quotechar, quoting, doublequote, escapechar, comment, encoding, encoding_errors, dialect, on_bad_lines, delim_whitespace, low_memory, memory_map, float_precision, storage_options, dtype_backend)
   1013 kwds_defaults = _refine_defaults_read(
   1014     dialect,
   1015     delimiter,
   (...)
   1022     dtype_backend=dtype_backend,
   1023 )
   1024 kwds.update(kwds_defaults)
-> 1026 return _read(filepath_or_buffer, kwds)

File ~\anaconda3\Lib\site-packages\pandas\io\parsers\readers.py:620, in _read(filepath_or_buffer, kwds)
    617 _validate_names(kwds.get("names", None))
    619 # Create the parser.
--> 620 parser = TextFileReader(filepath_or_buffer, **kwds)
    622 if chunksize or iterator:
    623     return parser

File ~\anaconda3\Lib\site-packages\pandas\io\parsers\readers.py:1620, in TextFileReader.__init__(self, f, engine, **kwds)
   1617     self.options["has_index_names"] = kwds["has_index_names"]
   1619 self.handles: IOHandles | None = None
-> 1620 self._engine = self._make_engine(f, self.engine)

File ~\anaconda3\Lib\site-packages\pandas\io\parsers\readers.py:1880, in TextFileReader._make_engine(self, f, engine)
   1878     if "b" not in mode:
   1879         mode += "b"
-> 1880 self.handles = get_handle(
   1881     f,
   1882     mode,
   1883     encoding=self.options.get("encoding", None),
   1884     compression=self.options.get("compression", None),
   1885     memory_map=self.options.get("memory_map", False),
   1886     is_text=is_text,
   1887     errors=self.options.get("encoding_errors", "strict"),
   1888     storage_options=self.options.get("storage_options", None),
   1889 )
   1890 assert self.handles is not None
   1891 f = self.handles.handle

File ~\anaconda3\Lib\site-packages\pandas\io\common.py:873, in get_handle(path_or_buf, mode, encoding, compression, memory_map,
is_text, errors, storage_options)
    868 elif isinstance(handle, str):
    869     # Check whether the filename is to be opened in binary mode.
    870     # Binary mode does not support 'encoding' and 'newline'.
    871     if ioargs.encoding and "b" not in ioargs.mode:
    872         # Encoding
--> 873         handle = open(
    874             handle,
    875             ioargs.mode,
    876             encoding=ioargs.encoding,
    877             errors=errors,
    878             newline="",
    879         )
    880     else:
    881         # Binary mode
    882         handle = open(handle, ioargs.mode)

FileNotFoundError: [Errno 2] No such file or directory: 'C:\\Users\\santo\\OneDrive\\Desktop\\Table_to_join_202511181007.csv'
```

In [ ]:

```
sql2
```

In [ ]:

```
```

In [18]:

```
pd.concat([sql, sql ])['Destination'].drop_duplicates()
```

Out[18]:

```
0     No Urgent Place
13               Home
16               Work
Name: Destination, dtype: object
```

In [20]:

```
sql[sql['passanger'] == 'Alone'][['Destination', 'passanger']]
```

Out[20]:

	Destination	passanger
0	No Urgent Place	Alone
13	Home	Alone
14	Home	Alone
15	Home	Alone
16	Work	Alone
...	...	...
12676	Home	Alone
12680	Work	Alone
12681	Work	Alone
12682	Work	Alone
12683	Work	Alone

7305 rows × 2 columns

In [21]:

```
sql[sql['weather'].str.startswith('Sun')]
```

Out[21]:

	Destination	passanger	weather	temperature	time	coupon	expiration	gender	age	maritalStatus	...	CarryAway	Rest
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	Unmarried partner	...	NaN	
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	Unmarried partner	...	NaN	
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female	21	Unmarried partner	...	NaN	
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female	21	Unmarried partner	...	NaN	
...	...	...	...	...	...	...	...	...	...	...	...	...	
12673	Home	Alone	Sunny	30	6PM	Carry out & Take away	1d	Male	26	Single	...	1~3	
12676	Home	Alone	Sunny	80	6PM	Restaurant(20-50)	1d	Male	26	Single	...	1~3	
12677	Home	Partner	Sunny	30	6PM	Restaurant(<20)	1d	Male	26	Single	...	1~3	
12678	Home	Partner	Sunny	30	10PM	Restaurant(<20)	2h	Male	26	Single	...	1~3	
12683	Work	Alone	Sunny	80	7AM	Restaurant(20-50)	2h	Male	26	Single	...	1~3	

10069 rows × 27 columns

In [22]:

```
sql[(sql['temperature'] >= 29) & (sql['temperature'] <= 75)]['temperature'].unique()
```

Out[22]:

```
array([55, 30])
```

In [23]:

```
sql[sql['occupation'].isin(['Sales & Related', 'Management'])][['occupation']]
```

Out[23]:

	occupation
193	Sales & Related
194	Sales & Related
195	Sales & Related
196	Sales & Related
197	Sales & Related
...	...
12679	Sales & Related
12680	Sales & Related
12681	Sales & Related
12682	Sales & Related
12683	Sales & Related

1931 rows × 1 columns

In [ ]: